Deep Political Leaning Analysis System (DPLAS)

2016 Spring - Web Retrieval and Mining - Final Project

林祐萱
B03902055
NTU CSIE
b03902055@ntu.edu.tw

蔡尚佑
B03902068
NTU CSIE
b03902068@ntu.edu.tw

林書瑾
B03902078
NTU CSIE
b03902078@ntu.edu.tw

陳力
B03902083
NTU CSIE
b03902083@ntu.edu.tw

1. INTRODUCTION

The main objective is to analyze a user's political leaning based on his posts. With vector space model, we treat query (user's posts) and documents (political parties' posts) as vectors. Then we can analyze user's political leaning with distance or similarity between their vectors.

Another important assumption is that posts about different topic may come from different language model. If we treat them as a large vector, it's unreasonable that a post can contain various topics. As a result, we will classify documents and user's query into different topics, creating a vector space model in each topic.

2. IMPLEMENTATION

The whole algorithm is basically a multi-class classification algorithm, with the political parties as classes. In most of time, text classification problem is simply applied with some linear classification techniques e.g. linear SVM. But here we think that since each document has its own topic like talking about gay-marriage.

The assumption we make here is that documents about different topic may come from different language model. So what we do first is to partition our documents into different topics by clustering algorithm. Then we train classifiers for each different topic base on the documents of the topic.

For a query (a user) , we treat it as a set of documents. Then we classify each document of that set (user) , aggregate the result and turns it into the result for that user.

2.1 Cluster Data Into Different Topics

The first job we have to do is to cluster data into different topics. We use *K-Means Clustering* to separate them into 12 different topics. The motivation of clustering data is: for documents with various topics, we assume they are generated from different language models.

For instance, a document about homosexuality is irrelative to capital punishment, and we'll assume it is generated from homosexuality model.

2.2 Build Bigram Vector Space Model

For each topic, we build a bigram vector space model based on data in the cluster. Thus, for a new document in this topic, we can measure it as a vector.

This model is based on *TF-IDF* and *Okapi BM25* smoothing method.

- $TF_{doc}(term) = \log(\frac{total\ docs - docfreq(term) + 0.5}{docfreq(term) + 0.5})$
- $IDF_{doc}(term) = \frac{c(term,doc)(k+1)}{c(term,doc)+k(1-b+b\times\frac{doc}{avgdoclen})}$
- $TFIDF_{doc}(term) = TF_{doc}(term) \times IDF_{doc}(term)$

where $c(term, doc)$ is the term frequency in document doc.

In this project, the parameters (k, b) are $(1.6, 0.75)$.

2.3 Measure Posts From Known Parties

For a post from a known party, we first determine which topics it discusses about, and measure it in the corresponding vector space model. This vector will be labeled to the specific party.

As a result, there are lots of labeled vectors in each topic model. They will be used to analyze user's political leanings.

Table 1: Result

Name	Party	Result(國民黨, 民進黨, 時代力量, 親民黨, 綠黨, 社民黨)
國民黨李大砲李新	國民黨	**0.357**,0.053,0.185,0.125,0.126,0.155
民主進步黨	民進黨	**0.298**,0.065,0.218,0.131,0.131,0.156
柯文哲	無黨籍	**0.260**,0.075,0.244,0.129,0.129,0.162
蕭美琴	民進黨	0.236,0.118,**0.238**,0.126,0.127,0.155
連勝文	國民黨	**0.366**,0.035,0.234,0.116,0.116,0.133
邱毅『談天論地話縱橫』	新黨（親民-國民黨）	0.174,0.188,**0.218**,0.130,0.131,0.159

2.4 Analyze User Political Leanings

To analyze a user's political leaning, we first have to extract the user's information. Here we treat each user as a set of vectors $\{d\}$, where each vector representing a post/article/document he wrote.

Then we use our algorithm above to estimate the probability $Pr(x$ is similar to posts of political party $y \| x, \theta)$ of each post it wrote. But it remains an issue what is the political leaning of that user given his post's political bias ?

Here we just average all political bias of its posts and treat it as the political leaning of that user. Next, we apply *K-Nearest Neighbors* algorithm to find k-nearest vectors (*logistic regression** is also used) with *cosine similarity*. Finally, we make k-nearest vectors to "vote" political leanings of user's posts.

*logistic regression: L2-regularized logistic regression, Fan, Rong-En, et al. "LIBLINEAR: A library for large linear classification." Journal of machine learning research 9.Aug (2008): 1871-1874. Appendix A.2

2.5 Active Learning

One more issue remains : what if that user don't post a lot?

Here we use an active-learning-flavored idea : provide several posts from out data set and make the user to choose which it identify with. The post the user choose can be seen as the post it wrote. And the extended user posts set is then applied with our algorithm to figure out the political leaning.

3. RESULT

The ground truth of a person's political leanings is hard to define. Hence, we estimate legislators' political leanings of different known parties. The results shown in *table 1* are estimated probability of having political leaning similar to some party. The parties used to build model are 國民黨, 民進黨, 時代力量, 親民黨, 綠黨, 社會民主黨.

For active learning, user's choice is usually determinative bacause the document is labeled to a political party already. As a result, when the input data is insufficient, this active-learning-flavored model works well.

4. CONCLUSION

In most cases, if the user input is sufficient and highly related to politics, the result is very precise. However, if user tends to post neutral and ambiguous remarks, or the input is insufficient, the precision will be limited. That is, the result is highly sensitive to user's input data. When user's posts are unbiased, there are too many documents considered similar.

Users' choice of documents is determinative when input data is not enough. Nevertheless, users' preferred documents may be not representative of his opinion. To summarize, the quantity of the user's posts are more important than his choice, which reveals his "true opinion" to some degree.

5. WORK DISTRIBUTION

- 陳力: model design , data clustering , website data crawling
- 林祐萱: demo website , FB posts data crawling , inverted file , model design
- 林書瑾: inverted file, topic VSM, *Liberty Times* news crawling
- 蔡尚佑: PTT Data crawling

6. DOCUMENT SET

We crawl data from Liberty Times news(自由時報), UDN news(聯合報), Facebook Pages, PTT Gossiping, political parties and NGOs. They can be downloaded from https://goo.gl/69bUCH